



AccessVia Web dSignShop Hardware Specification Guide

April 29, 2014

Table of Contents

AccessVia Web dSignShop Hardware Specification Guide	1
Table of Contents	1
Planning Ahead	2
Related Reference documents:	2
Minimum System Requirements	2
Database Platform System Requirements	2
System Load/Capacity Factors	2
Factors for Estimating Hardware Requirements	3
Application Server Types	4
Recommended Hardware Configurations	4
Server Tiers and Load Balancing	5
IIS Worker Processes and Processor Utilization	6
Hardware Failover and Redundancy	6
Production, Staging, and Development Server Environments	6
Scaling up the Environment Incrementally	7
Virtualized Hardware	7
64 Bit Processing Support	8
Database Server Environment	8
Estimating Actual Hardware Requirements	10
Determining Capacity Limits	10
Hardware Requirements Planning Worksheet	10

Planning Ahead

Planning ahead is vital for an AccessVia Web dSignShop system that performs at its best and fits the way your company does business. This guide discusses use considerations and the hardware components you'll need to build your Web dSignShop system.

Many factors can impact the actual hardware requirements for Web dSignShop. This document identifies many of those factors which should be taken into consideration when determining appropriate infrastructure requirements.

Often you'll have several options to choose from in terms of the type of hardware in your system and how you arrange the components together. Each combination of hardware options will differ in overall performance and price. Of course, the ultimate goal is to select those components that will perform best in your company's environment at an optimum price and minimum operating cost.

Related Reference documents:

Below are some additional reference documents which should be used in conjunction with this document.

- AccessVia Publishing Platform System Requirements
- AccessVia Web dSignShop System Diagram
- AccessVia Double Load Balancer Diagram
- AccessVia Scalability and Performance Guide
- AccessVia Staging and Production Environment Diagram

Minimum System Requirements

This document is intended to help determine overall hardware requirements and server topography for the Web dSignShop application environment. The minimum system requirements for each Web dSignShop server and client workstations are listed in the *AccessVia Publishing Platform System Requirements* document.

Database Platform System Requirements

Web dSignShop components supports a variety of database environments outlined in the *AccessVia Publishing Platform System Requirements* document.

System Load/Capacity Factors

Choose hardware that will support your heaviest periods of system usage. If one part of system is set up to handle heavy user loads, the system will still slow down if other components aren't able to keep up. Keep in mind that the following factors can contribute to, or help relieve, your system load:

- Processing Power
- Memory
- Caching – how caching is being managed by application and web server
- Database Access Speed
- Network Speed between end users and application servers

Factors for Estimating Hardware Requirements

There are many factors to take into consideration when estimating the total hardware requirements for Web dSignShop environment. All of these factors can affect your hardware requirements.

Peak Load Factors:

The key factor is the peak system load you anticipate supporting.

To calculate your peak load, you must pinpoint which hours of the day — and which days of the week, month, or year — generate the most signage. Along with this, determine the total number of store users that will be using the system at any one time. Typically, load peaks occur when a series of batches related to a large store promotion event are made available to stores for printing.

- Number of stores
- Number of users simultaneously accessing the system
- Volume of signs/labels generated during peak event

Store Operations Processes and Use Patterns:

Operation processes/policies and user habits are important. For example, do your store employees like to print the night before the sale, on the morning of the sale, or both? Other operational factors that affect that load are:

- Application use patterns
 - Frequency of promotions and price change events which result in sign printing
 - Time of day when stores perform sign printing activities
 - Store/User distribution across time zones
 - Window of lead time for production
 - Other major publishing events such as store openings or resets.
- Type of application use (Browsing activity vs. Printing Activity)
 - Volume of Batch Printing Activity
 - Volume of Ad hoc sign creation
 - Volume of Hand held request
 - Volume of hand held print requests

Shared system resources factors:

Other system environment factors should be considered are:

- Availability of shared resources in use by other applications (Server and DB)
- Availability of data sources
- Bandwidth availability/capacity

Application Configuration Factors

Many application configuration options you choose which will also affect your hardware needs are:

- Features related to how a job is printed, such as
 - Partial batch reprinting
 - Calculating the total pages for a print job

- Complex collation
- Page range features.
- Data integration features, such as
 - The type of pricing model used
 - Side trips to point of sale systems for price look-ups
 - Handheld application integration
 - Complex data import and export
- Template Portfolio configuration issues, such as
 - The total number of templates
 - The complexity of the most commonly printed templates
 - Use of graphics in templates
 - Business rules (expressions) in templates
 - Advanced template design features such as layers.
- Other application options, such as
 - The number and nature of UI pages
 - Template edit screens and list grids
 - Cache settings
 - The number and variety of print devices used

Preferred UI Response Times

You will also want to consider your desired Web UI response times and printer speeds.

Application Server Types

There are two primary web application server functions performed within the Web dSignShop environment.

- 1) **Web Server** – A Web Server hosts the Web dSignShop user interface by processing users requests for browsing the site, selecting batches, items, and creating ad hoc signs, to be printed.
- 2) **Publishing Engine Server (or Print Engine Server)** – The Publishing Engine Server processes print requests by merging data, graphics, and formatting rules to generate and render print or display output.

Both the Web Server and Publishing Server will require access to the Database Server.

Recommended Hardware Configurations

The Web Server and Publishing Server can be installed on the same server or separate servers.

For high volume production environments AccessVia recommends having the Web Server and Publishing (or Print) Server installed on separate dedicated servers or server clusters.

For low volume environments a single server can handle both Web and publishing/print functions.

If you plan to have your Web dSignShop components share server space with each other or with other unrelated applications, be sure to test your system to ensure a proper load balance.

Each company's use cases will be different, but the following table shows our general server hosting recommendations based on volume of Web dSignShop system usage. If you expect your usage volume to be near the maximum of one of the categories, it is a good idea either to follow the recommendations for the next-higher category or to invest in higher-performance hardware.

<p>High — more than 400 stores.</p>	<ul style="list-style-type: none"> • 2 or more Web servers and 2 or more print servers behind load balancer. • 4-process (or better) database server. • Maximum tiering. • Excellent failover protection.
<p>Medium — 30 to 400 stores.</p>	<ul style="list-style-type: none"> • 1 or 2 Web servers and 1 or 2 Print servers behind load balancer or in cluster – or – • 2 or 3 combined Web/print servers behind load balancer or in cluster. • 2- to 4-process (or better) database server. • Minimal tiering. • Good to excellent failover protection
<p>Low — fewer than 30 stores.</p>	<ul style="list-style-type: none"> • 1 combined Web/print server; no load balancer. • 2- to 4-process (or better) database server. • No tiering • No failover protection.

Contact your AccessVia Consulting Engineer for recommendations based on your specific business needs.

Server Tiers and Load Balancing

Companies with high volumes of activity can increase throughput, performance, and uptime by placing multiple Web Servers and Publishing Servers within a load balancer or cluster server environment.

The Web dSignShop application architecture supports multiple load balancing methods.

Load balancers can be configured to distribute requests to servers within each cluster, depending on which server has the lightest load at the moment. A load balancer enables addition and removal of servers without impacting users.

The Web Servers should have session redirection, also called “sticky sessions.” Publishing Servers can use round-robin redirection. The AccessVia Publishing Engine and the IIS environment will return the appropriate HTTP headers to allow redirection away from downed or non-responding servers.

An AccessVia Application Engineer can provide more information about this topic.

IIS Worker Processes and Processor Utilization

AccessVia Web Servers can only utilize one IIS worker process on a single machine so you can only utilize one processor for serving/processing the Web dSignShop user interface.

AccessVia Publishing/Print Servers can utilize multiple IIS worker processes on a single server so it can utilize more than one processor for print processing.

For this reason it is recommended to utilize multi-processor servers for both web and print functionality to optimize throughput. This is shown in the AccessVia Single Load Balancer Diagram.

More information on this topic can be provided by an AccessVia Application Engineer.

Please see the following related diagrams:

- AccessVia Web dSignShop System Diagram
- AccessVia Single Load Balancer Diagram
- AccessVia Double Load Balancer Diagram

Hardware Failover and Redundancy

The Web and Publishing Engine servers can be clustered in a web server farm to provide failover capabilities for each of the separate machines. See Server Tiers and Load Balancing section of this document.

For large infrastructure environments it's best to use a load balancer with lower cost single or two-processor machines for added reliability and flexibility rather than invest in a higher cost higher capacity four-processor machine for the web server.

Please consult your system infrastructure administrator for recommendations on hardware configuration options for server specific failover capabilities, such as disk mirroring, redundant network connectivity etc.

Production, Staging, and Development Server Environments

Staging Server Environment Overview:

AccessVia strongly recommends that you set up a “staging server” environment as a clone of your Web dSignShop production site. The staging environment will be used for testing and validating functionality, configuration changes, and other system modifications before making them available in the production environment.

Staging Server Environment Options:

The typical production system uses multiple servers in a server farm behind a load balancer. The staging server environment normally consists of one or two machines, depending on how closely you wish to mirror your production environment configuration. If your production configuration uses maximum tiering with separate web and print servers, then your staging environment should have two machines, one for each tier. Otherwise a single machine is fine. If your production configuration uses maximum tiering with separate Web and print servers, it is possible to host both Web and print functions on the staging server, though this may limit your ability to fully test performance and scalability of the system.

Development Server Environment Options:

If desired, you can clone additional copies of the environment into a development environment using minimal hardware for isolated development efforts.

Staging Database Options:

A separate database is required for each environment. However, staging environments don't necessarily need to be on a dedicated database server.. A common approach is to utilize a shared database server for the staging or test databases. The staging database server's environment is essentially the same as the production system, though performance requirements can be scaled down. AccessVia highly recommends a consistent naming convention in which all database server and web server names indicate their function. For example, server names might include "DEV", "STG", and "PRD" for development, staging, and production.

See: *AccessVia Staging and Production Environment Diagram*

Scaling up the Environment Incrementally

The Web dSignShop architecture allows you to easily scale up your environment. The application can be installed on minimal hardware infrastructure for initial setup and testing and then ramped up to accommodate full production levels closer to rollout timeline.

Use this document to plan for your initial and future hardware needs.

It's a good idea to set up your first Web dSignShop installation as a test system on minimal hardware requirements. After configuration and functional testing of the application are underway you can then replicate the environment and begin to build out your production environment. You can continue to use this server as a functional staging environment after you've rolled out your production system.

When you're ready to move to a production system for daily use, acquire separate machines to host the database, Web, and Print Engine servers. If scalability testing predicts a heavier system load than one set of servers can handle, you may decide to purchase additional servers right away.

Later on, if you find your system load increasing, you can add more Web servers, Print Engine servers, or both. The type of component you add depends on which part of the system proves to be the most heavily-loaded during daily use.

You can also convert a Web server to a Publishing server and vice versa to allocate your hardware to the function which is most needed.

Virtualized Hardware

Many AccessVia clients are using virtual hardware platforms. Virtualized server platforms in use by AccessVia customers include VMWare, Xenserver, and Hyper-V.

There are many advantages to using Virtual server platforms in the environment. Virtual servers are easier to maintain, support, modify, and replicate making them an attractive option for Web and Publishing server clusters.

There are some limitations to using to virtual servers for Web dSignShop.

- Limited Throughput - Virtual environments in some cases are known to provide significantly less throughput than physical servers with the same specifications. To mitigate these issues you should overestimate the hardware requirements by about 30% to accommodate for this.
- Limited Graphic Display Driver Resolution - Some configurations of Web dSignShop require use of a high resolution display driver to render output. Some Virtual server platforms have limited

display rendering capabilities. To mitigate this you should ensure your virtual server environment has high resolution display driver capabilities.

Virtualized hardware can be especially helpful hosting development and staging environments. However note that different hardware configurations for staging vs. production environments is not ideal, due to inherent difficulty reproducing and resolving issues which may be seen in your production configuration.

64 Bit Processing Support

64-bit operating systems are supported by AccessVia, however, AccessVia products are compiled in 32-bit and so do not fully utilize 64-bit processing architecture.

You must use the 32-bit control panel ODBC manager to setup DSN connections.

Database Server Environment

Disk Storage: Storage Area Network (SAN) vs. Standalone

Smaller organizations may prefer to use standalone systems for their Web, print and database servers. For instance, a decently equipped dual-CPU, dual-core server with six drives in a RAID 5 configuration *may be* entirely sufficient for expected database needs – both traffic volume and in future data growth.

Your organization may prefer to house frequently accessed data — such as database files — on a Storage Area Network (SAN) system. These systems are exceptionally suited to highly transactional environments where fast reading/writing of data, data redundancy, and a relatively easy path for disk space growth are at a premium. SAN systems tend to be much more expensive than comparably equipped Network-Attached Storage (NAS) systems or standalone servers.

There is really no preference between these choices when it comes to installing and using AccessVia software solutions. Both approaches will work well but will have different failover and management capabilities.

Server Redundancy: Grid Cluster vs. Failover Cluster

AccessVia highly recommends data and database redundancy for all of our clients. There are essentially two approaches to database server redundancy.

Grid Cluster

The first approach is called “grid clustering.” In this scenario, a cluster of several database servers is attached to a shared disk array back-end – likely a Storage Area Network (SAN) -- where the data resides. Typically these servers are connected to the SAN using a fiber card and not a typical Ethernet card. When a client or application needs data, it calls the “cluster.”

In the grid model, any of the servers in the cluster can respond by retrieving data from the shared disk array and returning it to the client or application. As data requests increase, additional servers can be added to the cluster to handle the load. Oracle RAC (Real Application Clusters) uses this method. Each cluster server in the RAC setup requires its own software license, which can be costly depending on the number of licenses needed for the whole system. Oracle RAC is known for being somewhat complex to setup and maintain – especially for a smaller organizations.

Failover Cluster

The second approach is to have a single redundancy-equipped server, or “primary node,” handle your database needs. A similarly-equipped server is then configured as a “failover cluster node” that connects to the same shared disk array as the primary node. This shared disk array can be a SAN, NAS, or use

mirroring to insure that the database storage is accessible from all nodes. The primary and failover nodes are typically connected to one another via standard Ethernet for the “heartbeat” signal to pass between them. If the primary node fails for whatever reason, the failover node takes over in a matter of seconds, keeping perceived downtime to nearly zero. The primary node can be repaired (or replaced, if need be) and brought back online once again. Multiple nodes can be used in multiple failover scenarios, and secondary nodes can be either active or passive. Active nodes allow use of the secondary machine for other, unrelated tasks. The Web documents listed below describe many potential configurations.

Failover clustering is used with Microsoft SQL Server when database server redundancy is required. In this case, you do not need a separate license for the Microsoft SQL Server software running on the failover server, thus keeping costs lower.

Although it may cost less to install a Microsoft SQL Server system, your organization may already be using Oracle software or an Oracle RAC system. Please consult your IT department to determine the most cost-effective solution for your organization.

Note: Adding more servers to a failover cluster configuration does not boost system performance, because only one server at one time is handling all database requests. In a grid cluster configuration, on the other hand, adding cluster nodes will increase performance, because concurrent nodes have access to the database at the same time.

Information on Oracle RAC

http://en.wikipedia.org/wiki/Oracle_RAC

<http://www.oracle.com/technology/products/database/clustering/index.html>

Implementing & installing Oracle RAC

<http://www.puschitz.com/InstallingOracle10gRAC.shtml>

http://publib.boulder.ibm.com/infocenter/wpdoc/v510/index.jsp?topic=/com.ibm.wp.ent.doc/wpf/inst_oracle_rac.html

<http://docs.hp.com/en/T1859-90017/ch02s05.html>

<http://www.oracle.com/technology/pub/articles/haskins-rac-project-guide.html>

SQL Server Clustering

http://www.sql-server-performance.com/articles/clustering/clustering_basics_2005_p1.aspx

http://www.sql-server-performance.com/articles/clustering/cluster_sql_server_2005_p1.aspx

http://www.sql-server-performance.com/articles/clustering/clustering_best_practices_p1.aspx

Data Models: Master Data Model (MDM) vs. Standalone

In the Master Data Model (MDM), critical and highly-used data is held in a central repository that is accessed by many systems. For instance, your retail item data (pricing, description, SKU, etc.) is stored in a common database that is shared by your POS systems, e-commerce website, signage software, and so forth. When item data changes it will appear immediately across the other systems sharing the data. The downside, of course, is that any errors in item data are also likely to be reflected across all systems using the data. Also, any changes to your data structure must be rigorously tested across all systems.

In a “standalone” data model, individual systems control their own data for their own use. For instance, your POS and e-commerce systems might each have their own distinct database systems that house their own item and pricing data. Because the systems are independent of one another, changes made to the POS system’s item data will not be reflected in the e-commerce system, and vice-versa. As might be expected, keeping data synchronized in this scenario can be difficult. On the positive side, data and data structure changes made in one system do not adversely affect the other systems.

AccessVia products can perform well with either data model. Choose the model that best suits your needs. Depending on the degree of MDM integration, your performance will vary relative to a stand-alone model.

Estimating Actual Hardware Requirements

Hardware infrastructure requirements can be estimated by forecasting application use, operational patterns and configuration options.

An AccessVia Application Engineer is able to make recommendations for hardware infrastructure by answering questions in the Hardware Requirements Planning Worksheet below.

Determining Capacity Limits

In order to determine precise hardware capacity limits AccessVia recommends performing Scalability and Performance Load testing on a fully configured test environment in the client specific environment with considering client specific factors.

Hardware Requirements Planning Worksheet

AccessVia can provide a questionnaire help identify and capture factors which will help determine hardware requirements. Please see the AccessVia Infrastructure Planning Questionnaire.